R64-34 Analysis of Computing-Load Assignment in a Multi-Processor Computer—M. Aoki, G. Estrin and R. Mandell. (*Proc. 1963 Fall Joint Computer Conference*, pp. 147–160.)

Increases in computer speed are expensive! Certain speeds at present are clearly not attainable. Consequently, when the required computational rate exceeds that which is available, some form of parallel processing must be resorted to. This parallelism may be accomplished physically in a variety of distinguishable ways; however, these systems are structurally all alike, consisting of two (or more) processors coupled through a common supervisory control, each having access to all, or part, of the available memory. Such a two-processor computer system is considered in the paper under review, the two computers are assumed to be structurally identical except for their computational speed. The analysis presented is applicable only to rather restricted systems but it must be recognized that this work represents one of the first analytical treatments of such problems.

The problem to which the authors address themselves is that of analyzing certain procedures for dividing the total computation between the two processors. The essential feature which should be recognized is that computation time must, in general, be treated as a random variable (for example, the execution time for some arithmetic instructions depends on the particular numbers being operated upon). Consequently, it becomes difficult to distribute the computation load between the two processors in a way which guarantees that both processors complete their tasks simultaneously. As the authors point out, in the case where intermediate results must be passed back and forth between the processors, the randomness in the computation time may cause serious delays to both processors.

The principal result (which unfortunately is not spelled out early in the paper) shows that for a distribution of computation time which is uniform over some interval the performance depends directly upon the ratio σ/m of the distribution's standard deviation to its mean. This is a rather interesting and simple result; it must be recognized that it is applicable only to the specialized models of parallel computation considered in the paper. The performance measure R is defined as the ratio of the expected computation time for the two-processor system to the expected time for a single-processor system. The authors fail to give a clear definition of just *which* single-processor system they are using. Their results show that R is bounded from below by $\frac{1}{2}$ and that this lower bound may be achieved when $\sigma/m=0$. The interpretation of this result must be made with care since the comparison is biased in favor of the two-processor system, *i.e.*, one clearly expects superior behavior for a two-processor system as compared to a single-processor system when all computational speeds are the same. Indeed, for such systems, it may be shown that $\frac{1}{2}\leq R\leq 1$ since at worst one of the two processors is always idle (giving $R=1$), and at best both processors are always busy (giving $R=\frac{1}{2}$).

The models analyzed in the paper consider the case of a fixed size common buffer which the two processors share. This buffer is used to hold the intermediate results of the first processor for subsequent use by the second processor. Consequently, the first processor must sit idle whenever the buffer is full and wait until the second processor is ready to empty (use) a portion of the stored results. On the other hand, the second processor will be forced to cease work whenever it needs new data and finds the buffer empty. The two cases considered by the authors are 1) a buffer capable of holding the results of one "unit" of intermediate computation and 2) an infinite buffer size. The first case was analyzed using a discrete model as well as a continuous model; for the discrete model, a computer evaluation was performed for a specific bivariate interpolation problem and the resultant computation and delay times are presented. It would be interesting to consider the analogies between the models presented in this paper and certain models of queueing theory, in particular, cyclic queueing models.

This reviewer would like to call attention to a small error in Table I of the paper under review. Values for σ/m (for a positive random variable which is uniformly distributed) are listed and the value $\sigma/m=1$ is included. It is easy to show that $\sigma/m\leq 1/\sqrt{3}$ for such a distribution. For this maximum value of σ/m we find that the maximum value of R is $\frac{2}{3}$. Certain other specific errors have been found. The description of Model 1 is inconsistent; the corrected rule 2 should state that "PU1 may not begin its $i+2$nd unit until PU2 completes its ith unit." The statement preceding (12), Model 1, should read "Var$(a)=4$ Var(b)." Typographical errors exist in (1), Model 1, where b_1 should be b_i, and in Appendix I. In (12), Model 2, the arguments of f and g must be interchanged.

In summary, it may be said that a new class of rather interesting and useful problems has been considered in this paper. Although the results are obtained after considerable analytic work and are themselves rather limited, it must be realized that models of parallel computation have heretofore been sadly neglected. Consequently, this work, representing one of the first treatments of such problems, is recommended to those interested in multiprocessor computer systems.

Leonard Kleinrock
Dept. of Engrg.
University of California
Los Angeles, Calif.